

محاضرة 4: أساسيات Digital Image - العمليات الرياضية

المحتوى: ① array & matrix operations

② linear & non-linear operations

③ Arithmetic operations

④ Sets & Logic operations

Array & Matrix operations

تمثيل الصورة في شكل مصفوفة Matrix هو نفس 2D-Array

الفرق في العمليات لا تعامل Matrix أو 2D-Array

الفرق في الماتلاب لما يعمل $x = x \wedge x \Rightarrow \text{Array}$

$x = x \wedge x = \text{Matrix}$

في Array العمليات تعمل العملية على كل عنصر والمقابل له

في Matrix العمليات ضرب والفرج وكل حزمة زي ما هنا عارفين في المصفوفات

ماتلاب كوني

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

- Matrix operation \Rightarrow Multiplication

$$A \times B = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix}$$

ضرب مصفوفات زي ما هنا عارفين

- Array operation \Rightarrow Multiplication

$$A \times B = \begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} \\ a_{21}b_{21} & a_{22}b_{22} \end{bmatrix}$$

ضرب كل عنصر

مع A في المقابل

مع B

Linear & non-Linear operations

* يقول رانك العملية Linear (أي شرط متحقق يبقى Non-linear operation)

Additivity -

Homogeneity -

لنفرض أن $f(x, y)$ هي صورة H operation ونكتب $g(x, y)$ هي صورة H operation

$$* H[a f(x, y)] = a H[f(x, y)] \rightarrow \text{homogenous}$$

$$* H[a_i f_i(x, y) + a_j f_j(x, y)] = a_i H[f_i(x, y)] + a_j H[f_j(x, y)]$$

Additive

* معنى ال Homogeneity رانك من خارج معايا ترتيب العمليات ، يعني من خارج

عملت $a * f(x, y)$ اولو وبعد كده طبقت ال H أو عملت $a * H$ وطبقنا العملية واحدة على $f(x, y)$ (الناتج ثابت)

معنى ال Additivity ، اني لو بعدت العملية على صورتين مثلا ، من خارج لو طبقت الصورتين و عملت ال H أو لو عملت ال H عليهم واحدة واحدة و طبقت الناتج

أمثلة للتوضيح : * عملية التجميع عملية Linear

$$\sum [a_i f_i(x, y) + a_j f_j(x, y)]$$

$$= \sum [a_i f_i(x, y)] + \sum [a_j f_j(x, y)] \Rightarrow \text{Additivity}$$

$$= a_i \sum [f_i(x, y)] + a_j \sum [f_j(x, y)] \Rightarrow \text{Homogeneity}$$

* عملية الجدار ال Max لقيم ال pixel linear

$$\text{Max} \left\{ (1) \begin{bmatrix} 0 & 2 \\ 0 & 3 \end{bmatrix} + (-1) \begin{bmatrix} 6 & 5 \\ 4 & 7 \end{bmatrix} \right\}$$

$$= \text{Max} \left\{ \begin{bmatrix} 0-6 & 2-5 \\ 0-4 & 3-7 \end{bmatrix} \right\} = \text{Max} \left\{ \begin{bmatrix} -6 & -3 \\ -4 & -7 \end{bmatrix} \right\} = \boxed{-2}$$

$$\text{But: } (1) \text{Max} \left\{ \begin{bmatrix} 0 & 2 \\ 0 & 3 \end{bmatrix} \right\} + (-1) \text{Max} \left\{ \begin{bmatrix} 6 & 5 \\ 4 & 7 \end{bmatrix} \right\}$$

$$= \boxed{3-7 = -4 \neq -2} \quad \boxed{2}$$

So, Max is non-linear operation.

Arithmetic Operations

Array operations عمليات الحساب كلها (ضرب، جمع، طرح، مقسمة) بتتق Array operations
لكل عملية تطبيقات هتسوفها:

① Addition (الجمع)

- من التطبيقات المهمة عملية ال Noise reduction .
- الفكرة، انه احياناً بيفضل ان Sensors بتجيب Noise مع الصورة
- هلق كذا صورة لنفس المشهد وأعمل averaging ليهنم ، هلق في ال noise بتقل
- ال averaging هو أجمع الصور (عملية الجمع هتبقى Array operation) و بعد كده اقسيم على عدد الصور (كل بكسل مقسمة على عدد الصور)
- كل ما زادت الصور بقي عندي نتيجة أحسن في تقليل ال noise
- نوع ال noise اللي بيضع معاها ال averaging أكثر هي ال gaussian noise
- راجع سلايد [4.5] وصفحة 19 (بترقم لبرناج مش الورق)

② Subtraction (الطرح)

- من التطبيقات المهمة بيتبقى في إضاءة مثلا (الكشاف العيوب ومراقبة الجودة)
- أو في إلقاء التحفة النسيجية ويتسوف السائل بيتحرك رازاي
- أو حتى في الخرائط لتوضيح الطرق مثلا أو صور الأقمار الصناعية
- * فكرة المصانع في مراقبة الجودة مثلا: بيلكون عندي صورة أصل بمقارنة بيها ، اسوها
- ال Golden image وتكون مثلا لعينة سليمة من المنتج ، كل منتج بعدي هافدك صورة image
- و أطرحها من ال Golden image ، لو فيه عنيت فيها قطعة ناقصة مثلا هيبقى فيه
- قيم ال pixel في المنطقة دي والباقى هيبقى نفس باره.

* فكرة الترابط من الصور الصناعية (راجع سلايد [4.6])
 - هيكل صورة ، وبعد كده يافد الصورة دي بيقبل منها نسخة ويخلي ال bit رقم صفر
 (least significant bit) بيفر ، وي طرح لصورتيه من بعض ، هتوضحه
 المسارات أكثر .

* فكرة المجال البني (اراجع سلايد [4.7])
 - هيكل صورة بيسمى ال Mask
 - بعد ها ليقيم الفرق بين المادة مشددة بحيث يعرف بلفق live image
 - هيافد الصورة ال live image وي طرحها من ال Mask
 - يعمل عليها تنويه enhancement عشانه يقدر يشوف تحرك المادة في جسم الخلية
 اراجع صفحة 20 في ملف السنة اللي فاتت

③ Multiplication (الضرب)
 اراجع من ال slides [4.8] و [4.9] وم ملفات
 السنة اللي فاتت صفحة 20

* وارد جدا بعد أي عملية ، انه قيمة ال pixel تؤدي أعلى قيمة مسموحة
 أو تبقى بالسالب ، في الحالة دي يعمل normalize and rescale
 قبل ما أحفظ الصورة ، دال هيعملها Clipping
 - أولًا : لو عندي قيم سالبة ، هتشف أكثر قيم سالبة وأطرحها من كل
 ال Pixels وأطلع صورة $f_m(x, y)$

$$f_m(x, y) = f(x, y) - \min[f(x, y)]$$
 - ثانيًا ، هافد قيمة أعلى pixel عندي وأقسم كل ال pixels عليها
 هتدني قيم بين صفر و واحد ، أقربها بقى في أعلى intensity مسموحة
 عندي ، وهي L التي قلنا قبل كده

$$L = 2^k$$

$$f_s(x, y) = L \left[\frac{f_m(x, y)}{\max[f_m(x, y)]} \right]$$

هتدي صورة قيم ال pixel فيها [4] بين (0 , L)

$$f_1(x,y) = \begin{bmatrix} 0 & 50 & 250 \\ 7 & 35 & 70 \\ 10 & 40 & 0 \end{bmatrix}, f_2(x,y) = \begin{bmatrix} 10 & 90 & 30 \\ 0 & 80 & 70 \\ 5 & 4 & 33 \end{bmatrix} \text{ مثال}$$

$$f_3(x,y) = \begin{bmatrix} 20 & 10 & 0 \\ 30 & 255 & 70 \\ 1 & 40 & 10 \end{bmatrix}$$

$$K=8 \Rightarrow L=256$$

$$f(x,y) = f_1(x,y) + f_2(x,y) - f_3(x,y)$$

$$= \begin{bmatrix} -10 & 130 & 280 \\ -23 & -140 & 70 \\ 14 & 4 & 23 \end{bmatrix}$$

① هافد أصغر قيمة سالبة وهي -140 وأطر هافد كل الـ pixels

$$f_m(x,y) = \begin{bmatrix} 130 & 270 & 420 \\ 117 & 0 & 210 \\ 154 & 144 & 163 \end{bmatrix}$$

② صنف الخطوة الثانية على الترقيم، عندي أكبر قيمة هي 420 فهضيم كل

$$\frac{f_m(x,y)}{\text{Max}[f_m(x,y)]} = \begin{bmatrix} 0.3095 & 0.6429 & 1 \\ 0.2786 & 0 & 0.5 \\ 0.3667 & 0.3429 & 0.0548 \end{bmatrix} \text{ عليها}$$

هضيم في L التي هي 256 [الصغ تقرب في 255 = L-1]
نمشي زي ما المحاضرة قالت [ونقرب الناتج لعدد صحيحة

$$f_s(x,y) = L \left[\frac{f_m(x,y)}{\text{Max}(f_m(x,y))} \right] = \begin{bmatrix} 79 & 165 & 256 \\ 71 & 0 & 128 \\ 94 & 88 & 14 \end{bmatrix}$$

المفروض
أضري 255

Sets and Logic operations

* راجع سلايد [4.12] ، يعرف فيها يعني Set والعمليات التي يمكن عملها زي Union (\cup) و Intersection (\cap) و difference و Complement

* سلايد [4.13] بتوضح شكل العمليات دي

* نفس الكلام في سلايد [4.14] ، سلايد [4.15] بتوضح تنفيذ العمليات

دي على صور ، مكتوب على اليمين العملية اللي حصلت على كل صورة

* ال Logic operation عوضها على صورة Binary في سلايد [4.16] .

لو هنطبق ال Logic operation على صورة $K=8$ مثلا ، هتبقى العمليات

Bitwise operation (هتعمل على كل pixel على مستوى bit)

* سلايد [4.17] مش مهمت قوي ، المفروض عارف الكلام اللي فيها ، امراها وبس